

CLAIMS

1. A computer program product, tangibly embodied in an information carrier, for developing an application, the computer program product being operable to cause data processing apparatus to:
 - 5 receive a first data model in a first language, the data model being used to structure development objects;
 - generate a set of intermediate objects based on the first data model; and
 - based on the set of intermediate objects and a code template, generate an API to access the development objects.
- 10 2. The computer program product of claim 1, further comprising instructions to convert the first data model to a second data model in a second language, wherein the set of intermediate objects is based on the second data model.
3. The computer program product of claim 2, wherein the second language comprises XML.
- 15 4. The computer program product of claim 1, wherein the first language comprises UML.
5. The computer program product of claim 1, wherein the set of intermediate objects comprises Java objects.
6. The computer program product of claim 1, wherein the first language comprises a
20 customizable extension.
7. The computer program product of claim 5, wherein the customizable extension is used to implement an additional feature of the API.
8. The computer program product of claim 6, wherein the additional feature comprises an indication of a file border.

9. The computer program product of claim 1, wherein the API comprises a copy and paste operation.

10. A computer program product, tangibly embodied in an information carrier, for developing an application, the computer program product being operable to cause data processing apparatus to:

receive a first data model in a first language, the data model being used to structure development objects;

generate a set of intermediate objects based on the first data model; and

based on the set of intermediate objects and a schema template, generate an XML schema used to implement the development objects.

11. The computer program product of claim 1, further comprising instructions to convert the first data model to a second data model in a second language, wherein the set of intermediate objects is based on the second data model.

12. The computer program product of claim 11, wherein the second language comprises XML.

13. The computer program product of claim 10, wherein the first language comprises UML.

14. The computer program product of claim 10, wherein the set of intermediate objects comprises Java objects.

15. The computer program product of claim 10, wherein the XML schema includes a tree based on aggregation relationships in the first data model.

16. The computer program product of claim 10, wherein the XML schema includes a reference based on an association relationship in the first data model.

17. The computer program product of claim 10, wherein the XML schema includes a complex type extension based on an inheritance relationship in the first data model.

18. A computer program product, tangibly embodied in an information carrier, for developing an application, the computer program product being operable to cause data processing apparatus to:

receive a first data model;

derive an API based on the data model; and

use the API to perform operations on a development object.

19. The computer program product of claim 18, wherein the API comprises an interface layer, a proxy layer, and a state layer.

20. The computer program product of claim 18, wherein the operations comprise:

creating a new development object without an existing corresponding file as a transient object; and

modifying the transient object until the transient object is committed to a persistent file.

21. The computer program product of claim 20, further comprising instructions to destroy the transient object if a delete is requested before the transient object is committed to a persistent file.

22. The computer program product of claim 20, further comprising instructions to mark the persistent file as deleted if a delete is requested after the transient object is committed to a persistent file.